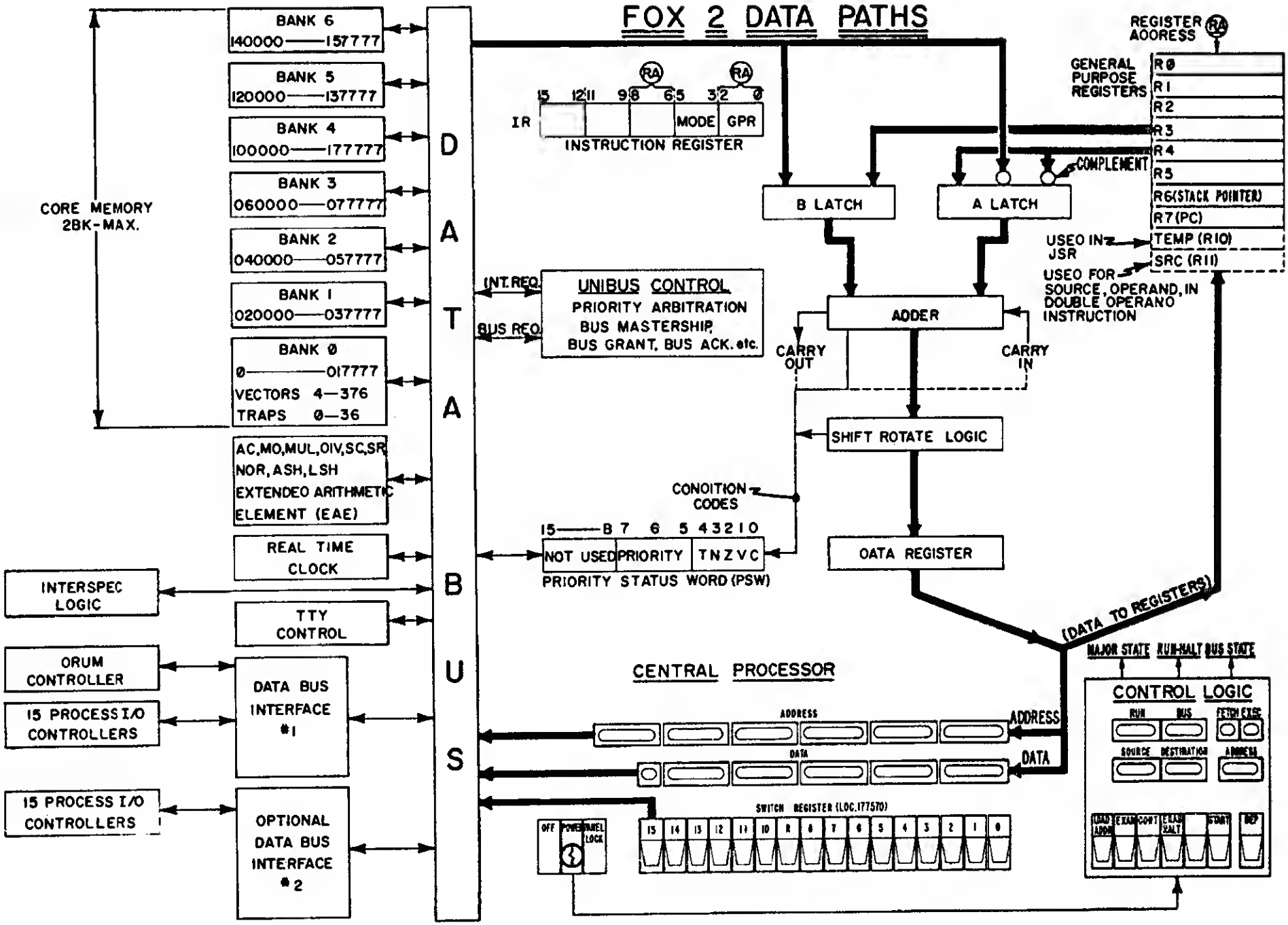
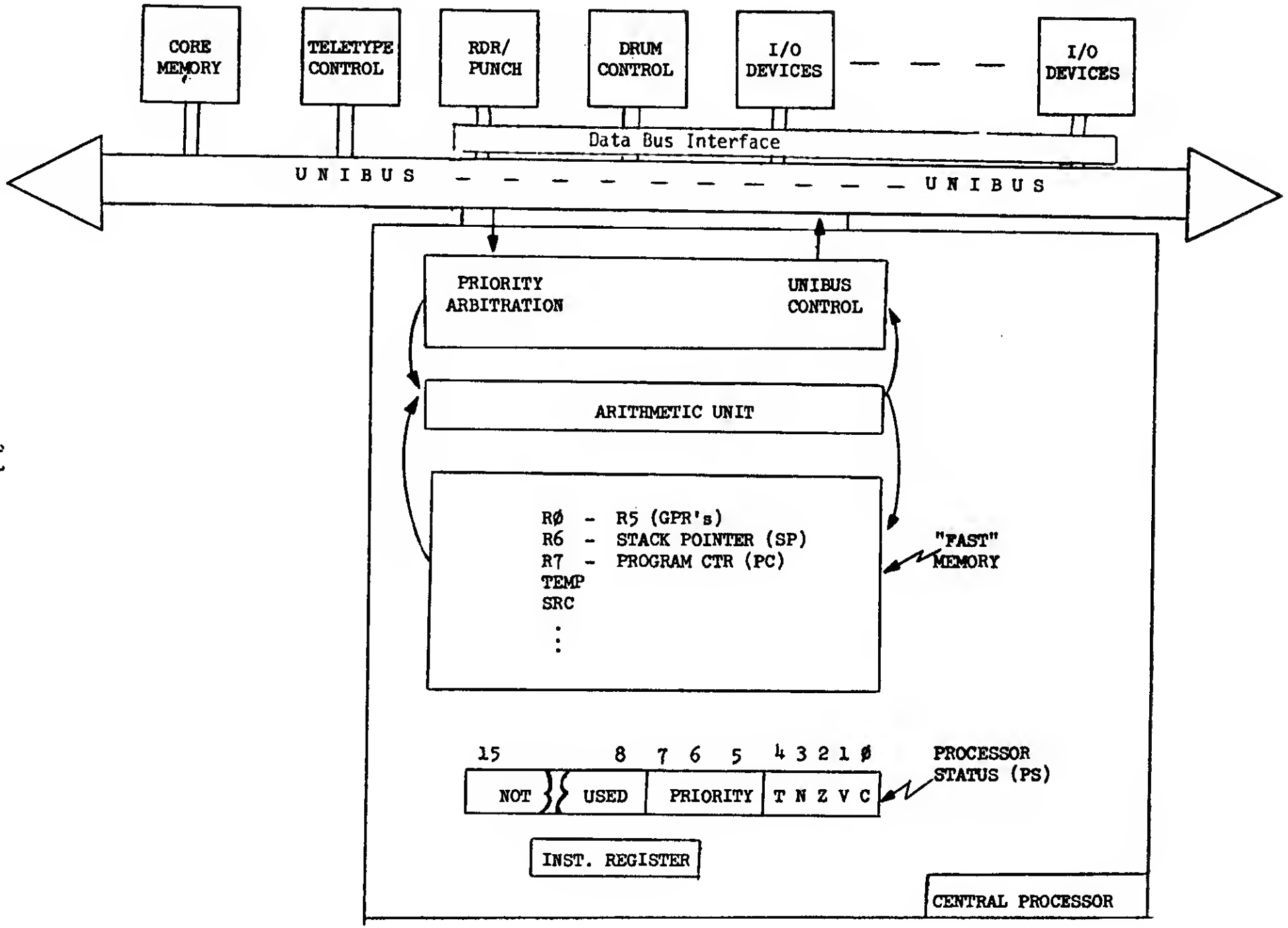
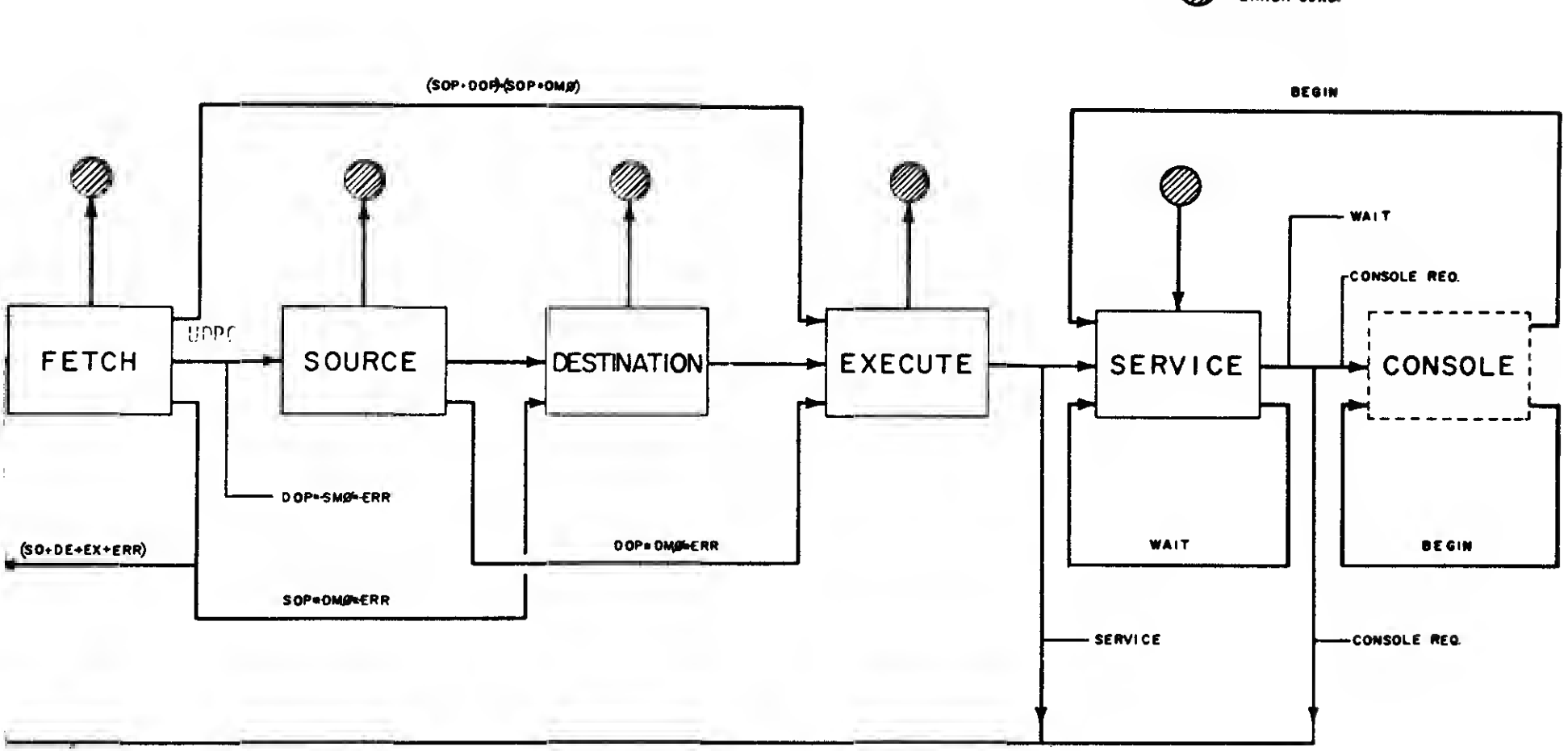

FOX 2 DATA PATHS
CORE MEMORY 28K-MAX.
BANK 6 140000—157777
BANK 5 120000—137777
BANK 4 100000—177777
BANK 3 060000—077777
BANK 2 040000—057777
BANK 1 020000—037777
BANK 0 0—017777
VECTORS 4—376
TRAPS 0—36
AC,MQ,MUL,OIV,SC,SR
NOR,ASH,LSH
EXTENDED ARITHMETIC
ELEMENT (EAE)
REAL TIME CLOCK
INTERSPEC LOGIC
TTY CONTROL
ORUM CONTROLLER
15 PROCESS I/O CONTROLLERS
DATA BUS INTERFACE #1
15 PROCESS I/O CONTROLLERS
OPTIONAL DATA BUS INTERFACE #2
DATA BUS
INSTRUCTION REGISTER
MODE
GPR
UNIBUS CONTROL
PRIORITY ARBITRATION
BUS MASTERSHIP
BUS GRANT, BUS ACK. etc.
INT.REQ.
BUS REQ.
B LATCH
A LATCH
COMPLEMENT
REGISTER ADDRESS
GENERAL PURPOSE REGISTERS
R0
R1
R2
R3
R4
R5
R6(STACK POINTER)
R7(PC)
TEMP (R10)
SRC (R11)
USED IN JSR
USED FOR SOURCE, OPERAND, IN DOUBLE OPERAND INSTRUCTION
ADDER
CARRY OUT
CARRY IN
SHIFT ROTATE LOGIC
CONDITION CODES
NOT USED
PRIORITY
T N Z V C
PRIORITY STATUS WORD (PSW)
DATA REGISTER
(DATA TO REGISTERS)
CENTRAL PROCESSOR
ADDRESS
DATA
SWITCH REGISTER (LOC. 177570)
CONTROL LOGIC
MAJOR STATE
RUN-HALT
BUS STATE
RUN
BUS
FETCH EXEC
SOURCE
DESTINATION
ADDRESS
START
DEP

FOX 2
BLOCK DIAGRAM



FOX 2 CPU
MAJOR STATE FLOW (SIMPLIFIED)



Example of Foxboro Module Addresses

MODULE	I/O CONN PLATE	IOT NO.	STATUS BIT	IOT ADDRESS	VECT ADDR	PRIORITY LEVEL
DBI	--	0	--	164000	--	-
DRUM	J14	1	1	164002	204	7
		2		164004		
		3		164006		
TAPE PUNCH	J1	4	2	164010	210	4
TAPE READER	J1	4	3	164010 //	214	4
DIG INPUT (FIELD)	J2	5	0	164012	200	5
(KYBD)	J2	5	4	164012	220	5
* ANALOG INPUT	J3	6	5	164014	224	6
* DIG DISPLAY	J4	7	6	164016	230	5
* DIG OUTPUT	J5	10	7	164020	234	5
* VAVLE CONTROL	J6	11	8	164022	240	5
* SYS SECURITY	J7	12	9	164024	244	6
* SETPOINT CONT	J8	13	10	164026	250	5
* PULSE COUNT	J9	14	11	164030	254	5
* PROCESS INTR	J10	15	12	164032	260	5
* SEL TYPER 1	J11	16	13	164034	264	4
* SEL TYPER 2	J12	17	14	164036	270	4

* OPTIONAL ADDR ASSIGNMENTS

SIMPLIFIED CORE MAP

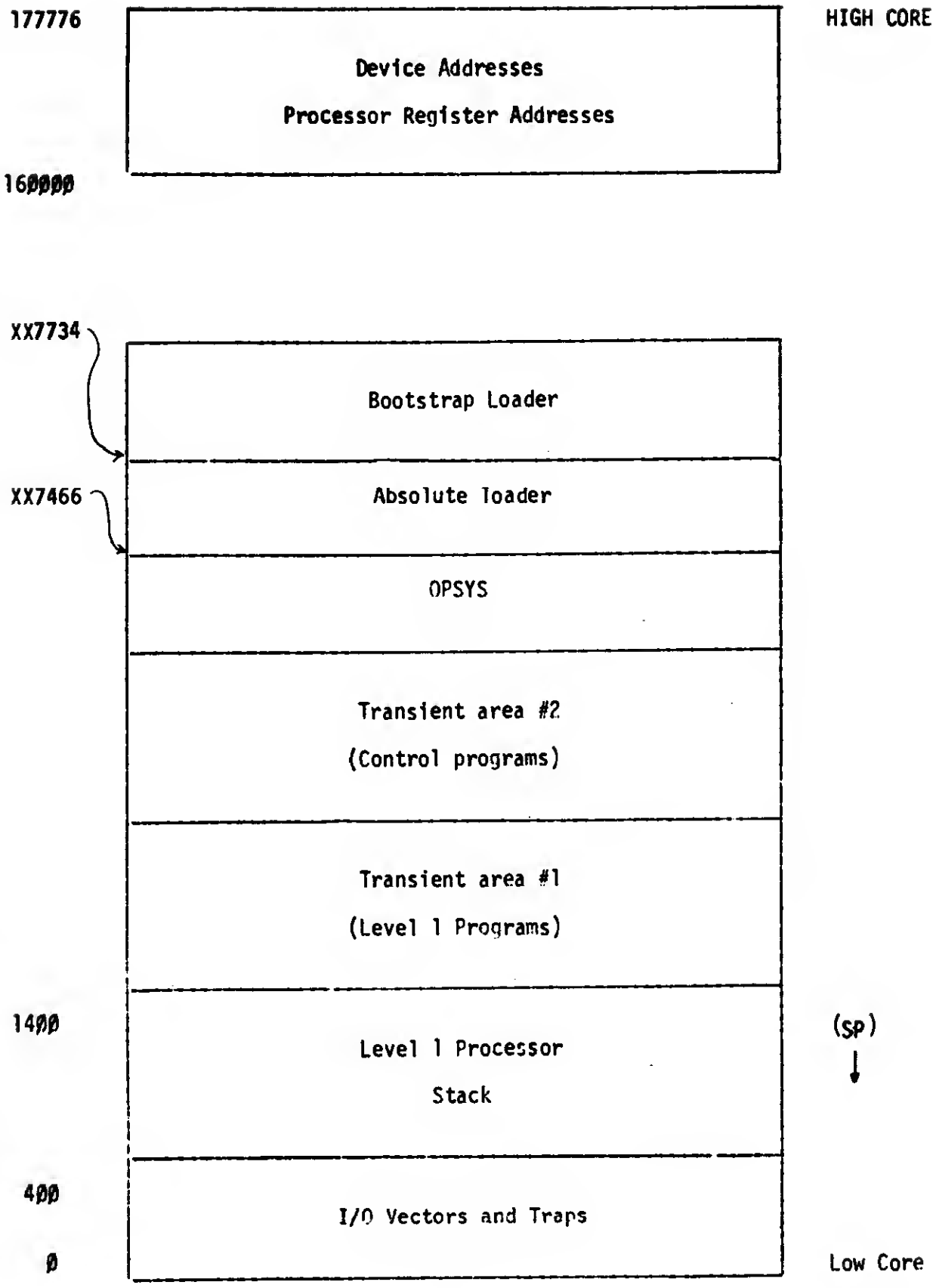


TABLE 9-1. Bootstrap Loader Instructions

a) Teletype Reader

OCTAL ADDRESS	OCTAL INSTRUCTION
xx7744	016701
xx7746	000026
xx7750	012702
xx7752	000352
xx7754	005211
xx7756	105711
xx7760	100376
xx7762	116162
xx7764	000002
xx7766	xx7400
xx7770	005267
xx7772	177756
xx7774	000765
xx7776	177560

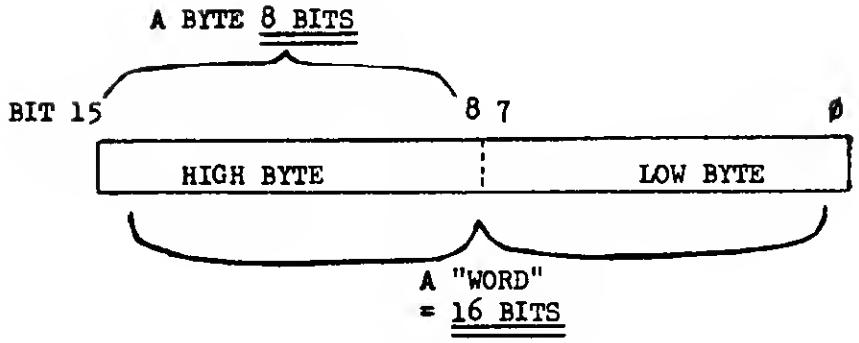
b) High Speed Reader

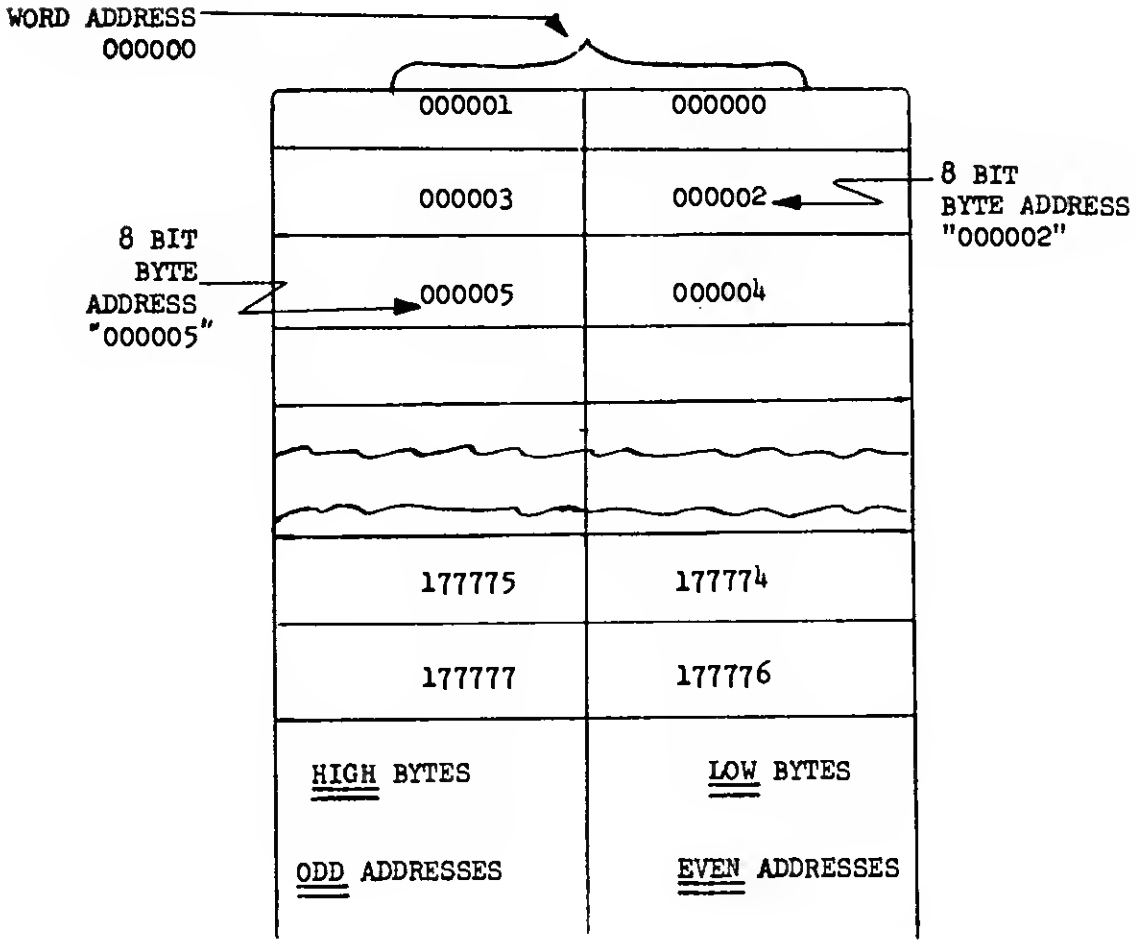
OCTAL ADDRESS	OCTAL INSTRUCTION
xx7734	016701
xx7736	000036
xx7740	012702
xx7742	000344
xx7744	112761
xx7746	000200
xx7750	000001
xx7752	132761
xx7754	000010
xx7756	177770
xx7760	001374
xx7762	011103
xx7764	110362
xx7766	xx7376
xx7770	005267
xx7772	177746
xx7774	000761
xx7776	164010

NOTE: The value of xx depends on core memory size, as follows:

Memory Size	Value of xx (octal)
8K	03
12K	05
16K	07
20K	11
24K	13
28K	15

ADDRESS STRUCTURE & TERMINOLOGY

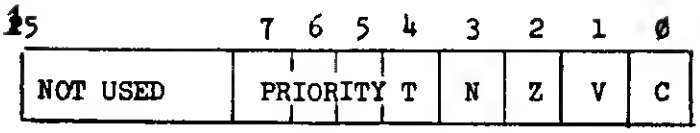




NOTES: 1. ONLY BYTES CAN HAVE ODD ADDRESSES

2. ALL INSTRUCTIONS OCCUPY A FULL WORD AT EVEN ADDRESSES, SO 2 MUST BE ADDED TO THE PC TO POINT TO NEXT INSTRUCTION WORD

PROCESSOR STATUS WORD



CONDITION CODE BITS

C BIT (BIT 0) SET IF CARRY FROM MSB
V BIT (BIT 1) SET IF ARITHMETIC OVERFLOW
Z BIT (BIT 2) SET IF RESULT ZERO
N BIT (BIT 3) SET IF RESULT NEGATIVE

TRACE TRAP

* T BIT (BIT 4) IF SET, CAUSES PROCESSOR TRAP (USED BY DEBUGGING PROGRAM)

PRIORITY

(BITS 5, 6, 7) SPECIFY CURRENT PRIORITY LEVEL OF PROCESSOR

* WHEN T BIT IS SET - COMPLETE ONE INSTRUCTION AND TRAP

GENERAL REGISTER ADDRESSING

MODE	DESCRIPTION	SYMBOLIC	ADDRESS CALCULATION PERFORMED
Ø	Register	R	(R) = Dperand
1	Register Deferred	@R or (R)	(R) → EA_f
2	*Auto-Increment	(R)+	(R) → EA_f; then (R)+ {1 or 2} → R (POP)
3	*Auto-Increment Deferred	@(R)+	(R) → EA_i, (EA_i) → EA_f; then (R) + 2 → R (POP)
4	*Auto-Decrement	-(R)	(R) - {1 or 2} → R (PUSH); then (R) → EA_f
5	*Auto-Decrement Deferred	@-(R)	(R) - 2 → R (PUSH); then (R) → EA_i, (EA_i) → EA_f
6	*Index	±X(R)	(NMW) + (R) → EA_f, where (NMW) = X
7	*Index, Deferred	@±X(R) or @(R)	(NMW) + (R) → EA_i, (EA_i) → EA_f, where (NMW) = X (NMW) + (R) → EA_i, (EA_i) → EA_f, where (NMW) = Ø

PC REGISTER ADDRESSING
NDTE: PC = %7

MODE	REGISTER	DESCRIPTION	SYMBOLIC	ADDRESS CALCULATION PERFORMED
2	7	Immediate	#N	(NMW) = Operand
3	7	Absolute	@#A	(NMW) → EA_f
6	7	Relative	A	(NMW) + UDPC → EA_f
7	7	Relative Deferred	@A	(NMW) + UDPC = EA_i, (EA_i) = EA_f

→ = Replaces the Contents of
EA_f = Final Effective Address
EA_i = Intermediate Effective Address

*All modes marked with an asterisk use a register's contents as an address, hence are really deferred modes. Modes 3, 5, and 7 are therefore doubly deferred modes.
NOTE: Deferred = Indirect.

R = Address of Register
(R) = Contents of Register
UDPC = PC After Being Updated Automatically

NMW = Next Memory Word

PUSH/POP RULES:
- By 1 if Byte Instruction
- By 2 if Word Instruction
- Except:
- Always 2 for R6 and R7
- Always 2 for Deferred Modes